

**REMARKS**

Claims 1-45 are pending in the Application, all of which have been rejected in the pending Office Action dated October 2, 2007. By this paper claims 1-4, 6, 7, 11, 16-19, 21, 26 and 31-45 have been amended. Independent claims 1, 16 and 31 have been amended for clarity and/or to further distinguish the claims over the cited references. Dependent claims 2-4, 6, 7, 11, 17-19, 21, 22, 26, and 32-45 have been amended to be consistent with the amended independent claims. No new matter has been added.

At paragraph 2 of the Office Action, the Examiner notes that the Information Disclosure Statement filed December 30, 2003 fails to comply with 37 CFR 1.97, 1.98 and MPEP §609. The Applicant agrees that the document the Examiner cites in the form PTO-892 is equivalent to the document the Applicant supplied with the form PTO-1449 on December 30, 2003.

While reviewing the application file, the Applicant discovered that an Information Disclosure Statement submitted in this application on April 8, 2004 was filed with a serial number having a typographical error (10/749,054, rather than the correct number 10/749,052). The original Information Disclosure Statement is attached. The referenced cited in the erroneously labeled Information Disclosure Statement are also cited herewith in a properly labeled Information Disclosure Statement.

At paragraph 4 of the Office Action, the Examiner rejects claims 3-15, 18-30 and 33-45 under 35 U.S.C. §112, second paragraph, as being indefinite. In particular, the Examiner asserts that claims 3, 18 and 33 do not clearly define the meaning of the term "child nodes" and their use in conjunction with register definitions. Accordingly, claim 1 has been amended to recite that "*the intermediate representation comprises nodes and links ... including one or more nodes representing a register definition,*" and claim 3 has been amended to recite that the child nodes are dependent from the "*one or more nodes relating to the ...register definitions*". Corresponding amendments

have been made to claims 16 & 18 and 31 & 33. In view of these amendments, the rejection is no longer proper and should be withdrawn.

Regarding the rejection of claim 40, the Applicant has amended that claim to depend from claim 36 as the Examiner suggests. Accordingly, that rejection should be withdrawn.

At paragraph 6 of the Office Action, the Examiner rejects claims 31-45 as relating to non-statutory subject matter. Accordingly, the Applicant amends claim 31 to recite “A computer apparatus ...”, which is clearly statutory subject-matter. Furthermore, taking into account the Examiner’s comments in paragraph 6 of the Office Action, further amendments have been made to claim 31 to require “a processor; a memory; and translator code stored in the memory and executed by the processor...”. In view of the amendments, these rejections are improper and should be withdrawn.

At paragraph 9 of the Office Action, the Examiner rejects rejects claims 1, 2, 16, 17, 31 and 32 under 35 USC §102(b) as anticipated by Aho1988. The Applicant traverses this rejection, and disagrees with the view that all of the features of these claims are disclosed by Aho1988. However, to expedite prosecution, the Applicant now respectfully submits that the independent claims as amended are clearly not anticipated by Aho1988 for at least the following reasons.

Firstly, Aho1988 only discloses methods of static compilation from a high-level source language to a machine-readable target code. See for example the statements on page 587. By contrast, amended claim 1 recites “*A method of performing dynamic binary translation to convert subject program code executable on a subject computing architecture into target code executed by a target computing system*”. This amendment is supported at least by paragraph [0027] of the description as filed, which explains that dynamic binary translation performs code conversion interleaved with execution of the generated code and that the translation converts a binary executable code (subject code) into binary executable code (target code) which is executed by the host (target) computing system. Thus, the applicant respectfully submits that the methods of static

compilation disclosed by Aho1988 do not anticipate the method of performing dynamic binary translation as required by claim 1.

Secondly, claim 1 recites:

- a.) *“grouping together a plurality of basic blocks of the subject program code to form a group block”*, (this amendment is supported at least by paragraph [0090] of the description as filed);
- b.) *“decoding the plurality of basic blocks of the subject program code in the group block”*;
- c.) *“generating an intermediate representation from said plurality of basic blocks of the subject program code in the group block, wherein the intermediate representation comprises nodes and links arranged as a directed acyclic graph representing expressions, calculations and operations performed by the subject program code, including one or more nodes representing a register definition by the subject program code”* (this amendment is supported at least by paragraph [0029] of the description as filed);
- d.) *“performing a partial dead code elimination optimization on said intermediate representation to generate an optimized intermediate representation, wherein the partial dead code elimination optimization traverses the intermediate representation to create a liveness analysis indicating when the register definition represented by the one or more nodes is a partially dead register definition which is live in one path and dead in another path through the group block”* (this amendment is supported at least by paragraphs [0157] and [0159] of the description as filed); and
- e.) *“generating target code from said optimized intermediate representation; and executing said target code on said target computing system”* (this amendment is supported at least by paragraph [0027] of the description as filed).

As to feature a.) above, Aho1988 discloses, on page 591, that “in the code optimizer, programs are represented as flow graphs, in which edges indicate the flow of control and nodes represent basic blocks.” Aho1988 discloses that “a basic block is a sequence of consecutive statements in which flow of control enters at the beginning and leaves at the end without halt or

possibility of branching except at the end" (Aho1988, page 528). Therefore, the Applicant respectfully submits that Aho1988 only discloses that code is optimized using basic blocks having one entry point, one exit point and in which a series of consecutive statements are defined. The Applicant respectfully submits that Aho1988 contains no disclosure or suggestion relating to this step of grouping together plural basic blocks to form a group block as in feature a.) of claim 1.

As to feature b.), it follows that, because Aho1988 does not anticipate the use of group blocks, the step of "decoding the plurality of basic blocks of the subject program code in the group block" is also not anticipated by Aho1988.

As to feature c.), the applicant respectfully submits that Aho1988 does not disclose or even suggest generating an intermediate representation from said plurality of basic blocks of the subject program code in the group block, not least because Aho1988 does not disclose such group blocks. Furthermore, the Examiner argues that the intermediate code disclosed on pages 598-590 and Figure 10.4 of Aho1988 is equivalent to the intermediate representation of claim 1. However, it is clear from Fig. 10.4 that the intermediate representation is a series of statements representing the code fragment of Fig 10.2 of Aho1988. By contrast, feature c.) of claim 1 recites that the intermediate representation is "a directed acyclic graph". Further, the directed acyclic graph of claim 1 represents "expressions, calculations and operations performed by the subject program code" and includes "one or more nodes representing a register definition by the subject program code". That is, the directed acyclic graph represents how the subject program code uses the hardware resources of the subject computing architecture. Thus, the applicant respectfully submits that feature c.) of claim 1 is not anticipated by Aho1988.

As to feature d.), it also follows that Aho1988 does not disclose or suggest the step of "performing a partial dead code elimination optimization on said intermediate representation to generate an optimized intermediate representation" as recited in claim 1. Here, the dead code elimination of Aho1988 is not equivalent to the partial dead code elimination required by the method claim 1. Dead code elimination involves identifying code that is "live at a point in a program" or "dead at that point" (Aho1988, page 595). No other state is considered. By contrast,

partial dead code optimization involves identifying code that is dead, code that is alive, and code that is partially dead. This partially dead code is dead in one branch of its destinations but live in the other of its branch destinations. It follows, therefore, that creating “a liveness analysis indicating when the register definition represented by the one or more nodes is a partially dead register definition which is live in one path and dead in another path through the group block” as recited in claim 1 is not disclosed or suggested by Aho1988.

Finally, as to feature e.), it follows that these steps are also not disclosed by Aho1988.

At paragraph 10 of the Office Action, the Examiner states that the prior art made of record and not relied upon is considered pertinent. The applicant submits that these documents are not relevant to claim 1, in particular because these documents are not relevant to dynamic binary translation as recited by claim 1. Therefore, the applicant respectfully submits that none of the cited documents offer any teaching, suggestion or motivation for the person of ordinary skill to consider adapting the teachings of Aho1988.

Thus, the Applicant respectfully submits that the method of claim 1 is neither anticipated nor rendered obvious by Aho1988 and the prior art made of record.

Corresponding arguments apply to the independent claims 16 and 31. Therefore, the applicant respectfully submits that these claims are also not anticipated or rendered obvious by the Aho1988 and the citations discussed above.

Filed herewith is a Request for a One-Month Extension of Time, which extends the statutory period for response to expire on March 17, 2008. Accordingly, Applicant respectfully submits that this response is being timely filed.

In view of the above amendment, applicant submits that the pending application is in condition for allowance and requests such action. No other fees are believed to be due in connection with the filing of this response. However, if a fee is due, please charge our Deposit

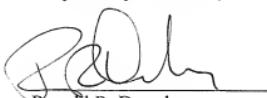
Application No. 10/749,052  
Amendment dated March 28, 2008  
Reply to Office Action of October 2, 2007

Docket No.: 1801270.00138US1

Account No. 08-0219, under Order No. 1801270.00138US1 from which the undersigned is authorized to draw.

Respectfully submitted,

Dated: March 28, 2008



Ronald R. Demsher  
Registration No.: 42,478  
Attorney for Applicant(s)

Wilmer Cutler Pickering Hale and Dorr LLP  
60 State Street  
Boston, Massachusetts 02109  
(617) 526-6000 (telephone)  
(617) 526-5000 (facsimile)